METHOD, SYSTEM AND STORAGE MEDIUM FOR PERFORMING ONLINE VALVE DIAGNOSTICS

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. provisional patent application serial number 60/447,598, filed February 14, 2003, the entire contents of which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

[0002] Due to wear, environment, age, misapplication, or manufacturing flaws, control valves used in the process industries can perform improperly or fail. When they perform improperly, the result is often lower yield of the process. If they fail it will often shut the process down. In a large refinery or chemical plant or power plant either of these outcomes can be very expensive. Plant operators have been interested in finding ways to detect problems in valve operation before the problem manifests itself in ways that hinder plant production.

[0003] Some common problems that occur in valves are: seat wear or damage, stem damage, stem corrosion, stem wear, packing wear, linkage problems/failures, actuator spring problems/failures, and actuator diaphragm failures. Many of these problems cannot be definitively identified except by disassembling the valve and examining the parts. This is a common practice in valve diagnostics, but is an expensive solution.

[0004] In recent years, operators have begun to infer many of the common problems from tests that they can make with the valve fully assembled and still in the process line (although when the process is not running such as during a plant shutdown). Valve "testers" have been built which attach sensors to the valves and make measurements. The most common measurements include positioning error, accuracy, response time, overshoot, spring range, and friction. To perform these tests, sensors must be attached to the valve and the valve must be stoked through much of its range. This prevents these tests from being performed during plant operation.

[0005] The use of data acquisition systems to gather valve operation data has been available for some time but has not been widely used for specific valve diagnostic applications. More often data acquisition system data has been used to evaluate overall process operation and in process tuning. Valve setpoint and valve position have been available using SCADA systems or the control system for some time.

[0006] More recently, some of the data acquisition has been moved to the valve itself in the form of smart positioners. Instead of the valve setpoint and position being measured from analog signals sent back to the control room, these measurements are made at the valve and sent back to the control room in digital form using one of several standard communications protocols. Some of the vendors of the smart positioners have included diagnostic applications in the firmware of the positioners allowing the positioner to control the valve to perform some diagnostic tests. These tests include response time tests, step response tests, actuator signatures, and friction analysis. With a few exceptions, however, these tests must still be performed when the process is not running.

[0007] There has been considerable interest in tests that can be performed when the process is still running. At least one vendor has built some tests in the firmware of their positioner to gather diagnostic information. This vendor requires the use of a special version of the positioner to work with software in the control room to perform these tests and can provide information about valves during the normal operation or during operation where the valve is artificially moved but only a small amount which would be tolerated by the running operation.

[0008] Thus, there is a need in the art for a method and system for performing valve diagnostics during normal valve operation.

BRIEF SUMMARY OF THE INVENTION

[0009] Embodiments of the invention include methods and systems for performing online valve diagnostics. Valve characteristics such as step response, friction and spring range are determined while the valve is operating in a process. Valve information is obtained while the valve operates in response to a control signal controlling a process while the valve operates through a series of gradual movements. Valve characteristics are then determined from the valve information.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] Figure 1 depicts a system for performing valve diagnostics in an embodiment of the invention.

[0011] Figures 2A-2D depict processing to derive a step response model.

[0012] Figure 3 is a flowchart of a process for determining model parameters.

[0013] Figures 4A-4D depict processing to determine friction.

[0014] Figures 5A-5B depict processing to determine spring range.

[0015] Figure 6 depicts a process for determining a lag generating a minimum error.

DETAILED DESCRIPTION

[0016] Figure 1 depicts a system for performing valve diagnostics in an embodiment of the invention. A process controller 10 generates a control signal used to establish valve position during normal operation of a process. A positioner 12 interfaces with a valve 14 in response to the control signal from process controller 10. Positioner 12 interprets an electrical control signal and controls an actuator to position valve 14. A pneumatic signal is sent to valve 14 to position the valve in response to the control signal and a feedback signal is provided to positioner 12. The feedback signal indicates the position of valve 14. Positioner 12 relies on a position feedback measurement to adjust the actuator.

[0017] Positioner 12 provides valve information to controller 16. The valve information includes the measured setpoint, measured valve position and actuator pressure. Valve position indicates the actual position of the stem where 0% means fully closed and 100% means full open. Actuator pressure indicates the air pressure in the actuator that is needed to position the valve. The air pressure is opposed by the springs inside the actuator and by any force of the process fluid flow and by friction.

[0018] The controller 16 executes a routine for performing valve diagnostics. The controller 16 may be a microprocessor-based controller and the routine may be embodied in a computer program executed by the controller. The controller 16 may be implemented using a variety of technologies such as an application specific integrated circuit (ASIC), field

programmable logic array (FPGA). Although, controller 16 is shown separate from valve 14, it is understood that controller 16 may be embedded within valve 14 as an integrated controller.

[0019] In operation, controller 16 monitors the valve information from valve 14 and performs valve diagnostics. During normal process operation, small changes in valve position occur due to changes in the setpoint sent to valve 14. The setpoint is the value of the signal converted to percent where 0% means closed and 100% means open. The valve 14 responds to the setpoint changes, moving the valve stem more open or closed. The controller 16 uses this data to compute characteristics such as error, offset, step response (including lag, response time (T86) and overshoot), oscillation, friction and spring range. As shown in Figure 1, these calculated characteristics may be used to provide alarms, reports and/or analysis to users. Embodiments of the invention include deducing the step response, the friction, and the spring range.

[0020] With respect to lag, the controller estimates the overall lag between the setpoint changes and the position response to the changes. The lag is determined by computing the error between the setpoint and the lagged position. The computed lag is the lag that yields the least square error. The lag is used in computing the error and offset.

[0021] Error and offset are common measurements made in existing valve diagnostic systems. A straightforward computation of error as the difference between the setpoint and position, however, has the undesirable property of being significantly affected by the response time. In embodiments of the invention, controller 16 defines error as the difference between the setpoint and the lagged position, and this undesirable property is minimized.

Controller 16 computes offset as the average of the error. Without the use of the lagged position, offset would have the same undesirable property.

[0022] Controller 16 also uses valve position spectral frequency to identify oscillation and limit cycling. As described in further detail herein, controller 16 uses standard Fourier techniques to obtain the spectrum.

[0023] A common diagnostic test is to change the setpoint abruptly and to see how the valve responds. Typically, the valve will not instantly follow the setpoint change but will take some time. The time it takes the valve to move 86% of the way from the starting point to the new setpoint it identified as T86 and is a common measurement. Once the valve reaches the new setpoint it may continue to move past the setpoint before returning to the correct position. This is called overshoot and is another common measurement.

[0024] During normal valve operation, the setpoint typically does not make any abrupt movements. Controller 16 may use two techniques to infer a step response from the gradual movements that occur during normal, in-process operation of valve 14. The first technique builds a model of valve response to setpoint changes. Using this model, controller 16 adjusts the model parameters until the prediction made by the model given the actual setpoint yields the minimum error between the prediction and the actual position. The model with these parameters can then be applied to a "step" in setpoint to create a graph of the step response in a form that those of ordinary skill in the art are familiar with. In one embodiment, the model used has two parameters, one that primarily affects the response time and one that primarily affects overshoot. The algorithm is described in further detail herein. From the result of this model, controller 16 predicts T86 and overshoot.

[0025] A second step response method involves the assumption that each small movement of the setpoint causes a series of responses (over time) from the valve 14. The valve position over time will be the sum of the responses from all of the individual changes in the setpoint. The time series response to each setpoint change is known as a wavelet. The wavelet that fits the data with least square error can be determined by solution of a series of linear equations. Applying the wavelet to a setpoint step will yield an estimate of the step response from which T86 and overshoot can be estimated. A detail derivation of the wavelet model is provided herein.

[0026] Friction is the force required to start the valve stem moving. For valves it is often measured in terms of the actuator pressure needed or as a percent of the actuator pressure needed. Friction is typically measured during an actuator signature by measuring the position hysteresis from increasing the actuator pressure and then decreasing the actuator pressure. The difference between the two resulting curves is proportional to the friction (stem friction plus actuator friction plus any deadband or other friction in the system).

[0027] When the valve in operation makes normal, small movements these movements move about small hysteresis loops. If the spring range of the valve is known, each of the samples of position versus pressure can be projected to the pressure axis using the slope of the spring range line. The spring range corresponds to the actuator pressures that cause the valve to be at the closed position and at the open position. These pressures are known as spring range because they are largely determined by the springs inside of the actuator.

[0028] This results in a "stretched normal" distribution which would be flat over a zone that represents the friction and tails off like a normal distribution outside of the flat area. The technique used in controller 16 projects the data to the pressure axis using the spring range slope and then finds the 90 percentile and 10 percentile values. The difference between these two values (of pressure) are an estimate of friction. This technique does not require any tracking of individual hysteresis loops and does not require any fitting of lines to the data and is thus very robust.

[0029] When the change in position throughout the sampled data is small, then errors in the assumed spring range will have almost no effect on the computed friction. When the position range is wider, it is possible to calculate the actual spring range to use.

[0030] Two additional methods of computing friction may be implemented. These methods do not require a pressure measurement and so can be used more broadly. The control system 10 typically makes very small corrections to the position. When these corrections are smaller than the friction (in percent of spring range) the valve will not actually respond to the changes. The control system 10 will then continue to change the setpoint until the valve finally responds. The change required in the setpoint until the valve finally moves is the friction percent. A first method uses the same modeling technique described above with respect to the step response. The model has a parameter that is the friction and the model computes no change in position until the integral of the changes to the setpoint in any direction exceeds the friction parameter, at which time the normal proportional-integral (PI) model is applied.

[0031] A second method computes the distribution of changes in the position versus the distribution of the changes in the setpoint. When a large peak exists in the position distribution that is at a higher position than the peaks in the setpoint, it represents a limit because of friction. Subtracting the setpoint change distribution from the position change distribution and finding the peak determines the friction.

[0032] Because of the hysteresis and because of inconsistencies in position versus pressure that occur during setpoint changes, fitting the position versus pressure data with a line to estimate spring range can be difficult. Embodiments of the invention implemented by controller 16 assume various spring ranges and project all of the sampled data to the pressure axis. The computed spring range is the spring range that minimizes the variance of the projected data. When the valve movement is more than 5% this method usually provides a good estimate of spring range.

[0033] Additional detail concerning the processing performed by controller 16 will now be described. Embodiments of the invention employ processing analogous to wavelet processing used in seismic exploration. Let $s = s_0, s_1, s_2, s_3, \ldots$ be the input (setpoint) signal expressed in position percent units and $p = p_0, p_1, p_2, p_3, \ldots$ be the valve position. Figure 2A illustrates the setpoint and position with respect to time. We assume that each change in the signal represents an impulse to the valve which responds with an extended output or wavelet $\omega = \omega_0, \omega_1, \omega_2, \ldots \omega_n$. Expressed mathematically,

$$\frac{dp}{dt} = \left(\frac{ds}{dt} \otimes \omega \right)$$

[0034] Integrating both sides we get

$$p = \int \left(\frac{ds}{dt} \otimes \omega \right) dt$$

[0035] Let $i = i_1, i_2, i_3, \ldots$ be the time derivative of the (input) signal and $o = o_1, o_2, o_3, \ldots$ be the time derivative of the (output) position. Figure 2B illustrates the derivative of setpoint and position. We can write out the convolution in the equation above as:

$$o_k = (i_k * \omega_0) + (i_{k-1} * \omega_1) + (i_{k-2} * \omega_s) + \ldots (i_{k-n} * \omega_n)$$

[0036] This has the form of a linear equation with independent variables $i_k, i_{k-1}, \ldots$, dependent variable o_k, and coefficients $\omega_0, \omega_1, \ldots \omega_n$. Since we have many samples to work with, we can find a least squares estimate of the coefficients – i.e. the set of coefficients that gives the least error of estimating o_k from the i_k using the formula.

[0037] The least square formulas can be formed by multiplying the equation by each independent variable and summing to form a set of n+1 linear equations:

$$\sum (o_k \cdot i_k) = \omega_0 \sum (i_k \cdot i_k) + \omega_1 \sum (i_k \cdot i_{k-1}) + \ldots \omega_n \sum (i_k \cdot i_{k-n})$$

$$\sum (o_k \cdot i_{k-1}) = \omega_0 \sum (i_{k-1} \cdot i_k) + \omega_1 \sum (i_{k-1} \cdot i_{k-1}) + \ldots \omega_n \sum (i_{k-1} \cdot i_{k-n})$$

$$\sum (o_k \cdot i_{k-2}) = \omega_0 \sum (i_{k-2} \cdot i_k) + \omega_1 \sum (i_{k-2} \cdot i_{k-1}) + \ldots \omega_n \sum (i_{k-2} \cdot i_{k-n})$$

$$\sum (o_k \cdot i_{k-n}) = \omega_0 \sum (i_{k-n} \cdot i_k) + \omega_1 \sum (i_{k-n} \cdot i_{k-1}) + \ldots \omega_n \sum (i_{k-n} \cdot i_{k-n})$$

[0038] If we use the symbols $C_{xy}(n)$ to be the cross product $\sum x_k \cdot y_{k-n}$ and $A_x(n)$ is the cross product (autocorrelation) $\sum x_k \cdot x_{k-n}$. These are the cross correlation and the autocorrelation functions. We can then write the equations as:

$$C_{oi}(0) = \omega_0 \, A_i(0) + \omega_1 \, A_i(1) + \omega_2 \, A_i(2) + \ldots \omega_n \, A_i(n)$$
$$C_{oi}(1) = \omega_0 \, A_i(1) + \omega_1 \, A_i(0) + \omega_2 \, A_i(1) + \ldots \omega_n \, A_i(n-1)$$
$$C_{oi}(2) = \omega_0 \, A_i(2) + \omega_1 \, A_i(1) + \omega_2 \, A_i(0) + \ldots \omega_n \, A_i(n-2)$$
$$C_{oi}(3) = \omega_0 \, A_i(3) + \omega_1 \, A_i(2) + \omega_2 \, A_i(1) + \ldots \omega_n \, A_i(n-3)$$

...

$$C_{oi}(n) = \omega_0 \, A_i(n) + \omega_1 \, A_i(n-1) + + \omega_2 \, A_i(n-2) \ldots \omega_n \, A_i(0)$$

[0039] Written in matrix form:

$$\begin{bmatrix} A_i(0) & A_i(1) & \ldots & A_i(n) \\ A_i(1) & A_i(0) & & A_i(n-1) \\ \ldots & & & \\ A_i(n) & A_i(n-1) & \ldots & A_i(0) \end{bmatrix} \begin{bmatrix} \omega_0 \\ \varpi_1 \\ \ldots \\ \omega_n \end{bmatrix} = \begin{bmatrix} C_{oi}(0) \\ C_{oi}(1) \\ \ldots \\ C_{oi}(n) \end{bmatrix}$$

[0040] This can be solved by common matrix methods and results in a wavelet such as that shown in Figure 2C. The controller 16 uses a Gauss Jordan elimination method with row swapping to solve for the wavelet.

[0041] The derivative of a step input is an impulse. To calculate a step response model from the wavelet, the wavelet is convolved with a single impulse, which corresponds to the wavelet multiplied by a constant. This convolution is shown in Figure 2D. The step response output (position) is the integral of the wavelet times the constant.

[0042] Another embodiment of the invention uses a proportional-integral (PI) model, although embodiments of the invention are not limited to this type of model. The model parameters are defined as follows.

$$Error_i = (Setpoint_{i-1} - Estimate_{i-1})$$

$$Estimate_i = Estimate_{i-1} + Error_i * P + SumError * I + Offset$$

$$SumError = SumError + Error_i$$

[0043] The value of offset is computed as described above (the average of the error) and variable SumError is initialized to 0. Controller 16 determines P and I so that the square of the error between variables Estimate and Position is minimized. Two techniques may be used to determine P and I in order to minimize the square of the error. The first technique works recursively by for each parameter, finds a local minimum of the error by adjusting the parameters that follow it. This technique is depicted in Figure 3 and is self explanatory.

[0044] A second technique works by creating a grid of N sections over the space of allowed values of each of the parameters. The error is calculated at each of the grid points. The grid section that contains the minimum error is selected and that grid is divided into N sections and the process is repeated until the lower limit of change is reached.

[0045] During normal operation of the valve, the setpoint will be changed slightly by the controller 10 to make fine adjustments. This will result in the positioner 12 making small adjustments in actuator pressure. But because of friction, the actuator pressure will increase or decrease some amount before the valve 14 actually moves. If pressure versus position is plotted, instead of following a line (representing the spring range), the points will follow a band around the spring range line, the width of the band representing the friction. Figure 4A depicts a plot of pressure versus position.

[0046] If the axis of the graph (pressure on the x axis, position on the y axis) is rotated so that the spring range line is vertical as shown in Figure 4B, then, ignoring effects

of process variations, the statistical distribution of the pressures will be approximately normal outside of the friction band and will be flat within the friction band as shown in Figure 4C. The controller 16 estimates friction by finding the pressures within which 80% of the points lie after the graph has been rotated. The difference between the 90 percentile pressure and the 10 percentile pressure, shown in Figure 4D, represents 2 times the friction. The reason for using the 10 percentile and 90 percentile (or other values such as 20 and 80) is that some of the valve movements, if rapid, will cause a lag between the pressure change and valve movement. This will cause the pressure-position point that was sampled to be outside of where it would normally be. Excluding the outer 20% of the points yields a better estimate of friction since the points left are probably not ones that were affected by the sudden movements.

[0047] Controller 16 also determines the spring range for valve 14. As described above, to perform the friction calculation, the axis of the pressure versus position graph is rotated so that the spring range line is vertical. This provides an approximate spring range that is used for friction calculation. It is possible, however, to calculate the spring range by rotating the axis if enough data is present. Figure 5A depicts the axis of the pressure versus position graph being rotated. To compute the spring range, controller 16 searches for a rotation value that makes the variance of the data points a minimum. Figure 5B depicts the data distribution for two different rotations of the axis, one having a lower variance. The average of the resulting pressure values is the lower spring range and the amount of rotation determines the slope (and thus the upper spring range).

[0048] The lag calculation is determined by finding the lag that results in the minimum summed square error from the setpoint. The values

$$Error_i = (Setpoint_i - Position_{i+lag})^2$$

$$SummedSquareError = \sum Error$$

are computed by controller 16 for each lag within the accepted range. The lag for which SummedSquareError is minimum is the computed lag. Figure 6 depicts a process for determining the lag that generates the minimum sum of squared error.

[0049] With respect to offset and error, controller 16 computes error after the lag has been determined as follows.

$$Error = ((\sum((Setpoint_i - Position_{i+lag})^2))/NumberOfPoints)^{1/2}$$

[0050] The offset is computed as the average of the raw error as follows.

$$Offset = (\sum(Setpoint_i - Position_{i+lag}))/NumberOfPoints$$

[0051] Embodiments of the invention provide for taking data with the process running, but not intervening at all by changing what the process controller does. In other words, controller 16 acts as a passive listener. If the plant engineer intervenes and causes small changes in the process, controller 16 produces better information, however, it is not required that the engineer make any changes. In fact, tests performed by controller 16 are made automatically on a scheduled basis without the engineer even being aware that the tests are being made.

[0052] While the invention has been described with reference to exemplary embodiments, it will be understood by those skilled in the art that various changes may be

made and equivalents may be substituted for elements thereof without departing from the essential scope thereof. Therefore, it is intended that the invention not be limited to the particular embodiments disclosed for carrying out the invention, but that the invention will include all embodiments falling within the scope of the appended claims.